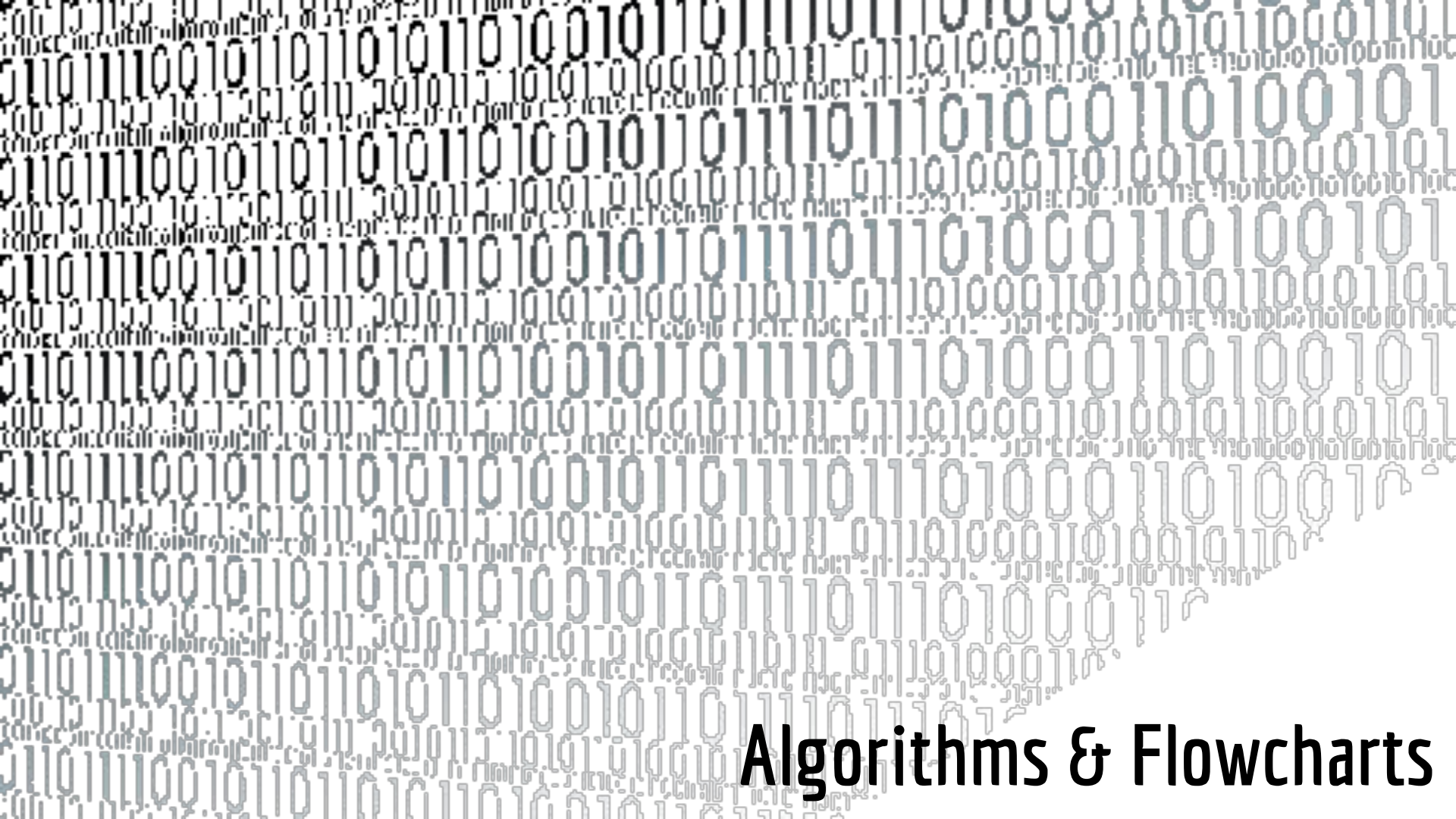


Algorithms & Flowcharts

Definition

- An algorithm is a sequence of steps used to solve a problem from beginning to end
- It is a step-by-step set of instructions

That is, the algorithm instructs, it does not necessarily provide solutions.

- Pseudocode: the incorporation of basic commands of syntax in the algorithm

Steps

- The steps in an algorithm will document the operations that are used to solve a problem
- The steps in an algorithm fall under the following four categories:

1. Output
2. Input
3. Process
4. Decisions

1) Output

- Any output statements
  - It is not necessary to document exactly what is outputted.
  - Simply document the purpose of the output.
  - For example:
    - Output title of program
    - Output description of program
    - Prompt for name
    - Output answer

2) Input

- Any information that is needed from the user to solve this problem.

- For example
  - **<u>Get</u> price**

3) Process

- Calculations needed to solve a problem
- You do not need the actual solutions or even formulas at this point.
  - You simply need to know what you need to calculate

- For example:
  - Calculate area

4. Decisions

- Any action that is based on a condition.
- Determines the flow of your solution
- For example:
  - Prompt for price
  - Get price
  - If price >= 4.00 then
    - Calculate tax
  - Calculate total cost

An example of an algorithm

Area of a triangle

- output title
- output program description
- prompt for base
- get base
- prompt for height
- get height
- calculate area of a triangle
- output area

Algorithm	Code
output title	print (“Area of Triangle”)
prompt for base	base = input (“Enter base”)
get base	base = float(base)
prompt for height	height = input (“Enter height”)
get height	height = float(height)
calculate area	area = base * height/2
output area	print (“Area of triangle is”, area)

What are Flowcharts?

- They are used to plan the process of the program before actually making it
- Used in designing and documenting the simple processes and programs
- Help in visualizing what is happening in the process
- Makes it easier to spot out flaws, bottlenecks or other less obvious problems
- Specific shapes show different commands

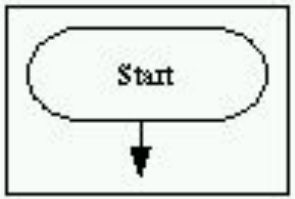
Flowchart elements

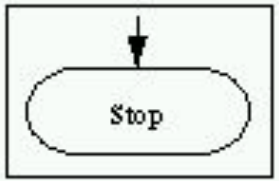
- We will be looking at five boxes used by programmers in flowcharts
  - Each box illustrates some type functionality in a program (such as input and output)

- Each box is connected using an arrow.
  - This arrow will show the direction of the flow from one box to the next

Terminal boxes

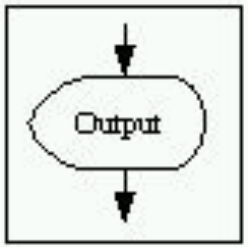
- These are the starting and ending points of your flowchart





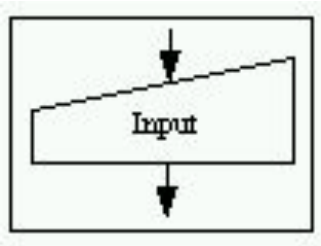
Output boxes

- Used to illustrate data being presented to the user.



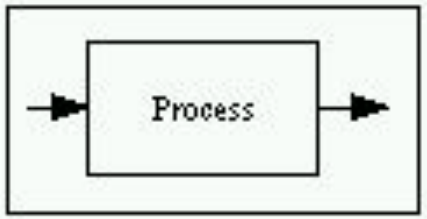
Input boxes

- Used to illustrate data being provided to the algorithm



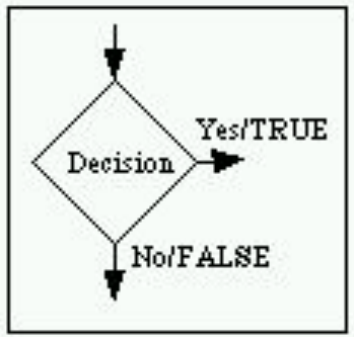
Process box

- Used to indicate where some sort of action takes place
  - For example: a change in value, form, or location of information



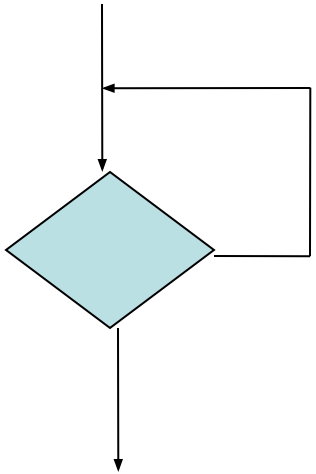
Decision/Condition box

- Used to allow for branching in algorithms
  - Information is checked and a decision is made as to how the flow of the program is to continue
  - Used to illustrate the ‘if’ statement



Repetition

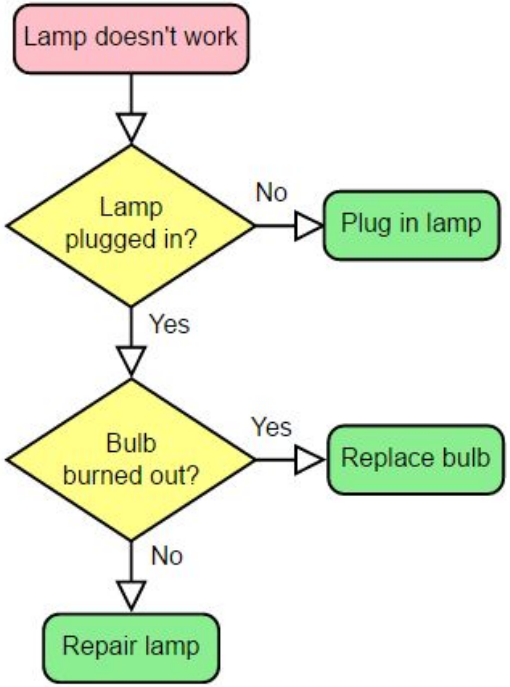
- Repetition also uses the decision box.
- If the condition is true then the arrow will branch off back to the point that is to be repeated

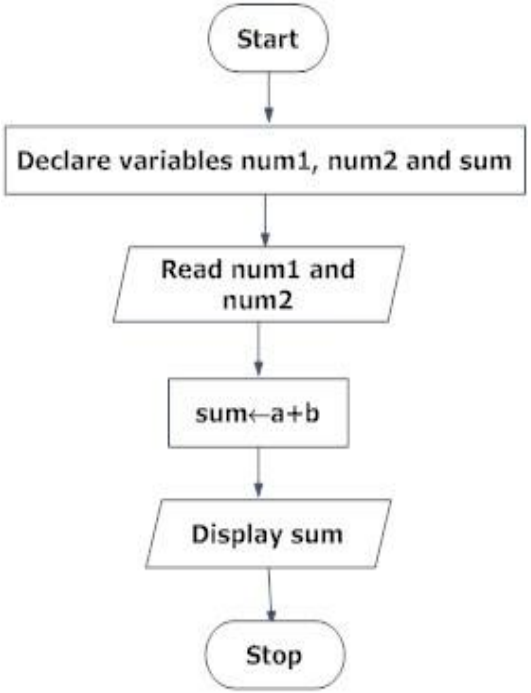


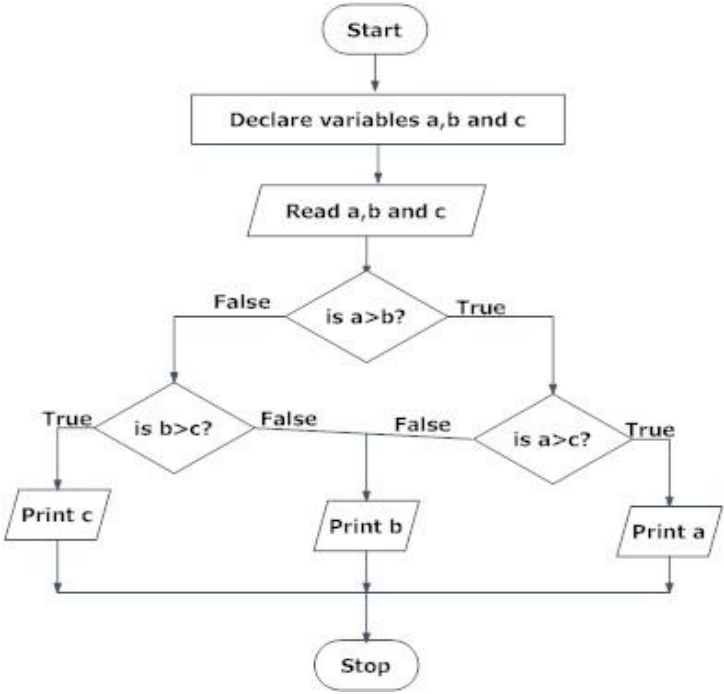
Flowcharts on Google Docs

- All flowchart boxes can be drawn using Google Docs
- Insert -> Drawing
- Click on the icon two positions to the right of the arrow and then click on shapes
- Towards the bottom, you will see flowchart elements

Examples







The parallelogram can be used for input or output as well but stick to the specific two symbols seen in previous slides.

Your Turn:

1. Using the shapes, make a flowchart of your day-to-day processes after you wake up in the morning.

    - Example: you wake up and check the time. If before 7 a.m., you take a shower, or else you just brush your teeth.

- Draw a flowchart for the following problem:

Write a program that prompts the user for a mark between 0 and 100. This program will determine if the mark is a pass (>=50) or a fail (<50) and output the appropriate message.